Multiple Random Walks to Uncover Short Paths in Power Law Networks

Bruno Ribeiro[1], Prithwish Basu[2], and Don Towsley[1]

[1]Computer Science Department
University of Massachusetts Amherst
{ribeiro, towsley}@cs.umass.edu

[2]Raytheon BBN Technologies
Cambridge, MA
pbasu@bbn.com

Abstract—Consider the following routing problem in the context of a large scale network G, with particular interest paid to power law networks, although our results do not assume a particular degree distribution. A small number of nodes want to exchange messages and are looking for short paths on G. These nodes do not have access to the topology of G but are allowed to crawl the network within a limited budget. Only crawlers whose sample paths cross are allowed to exchange topological information. In this work we study the use of random walks (RWs) to crawl G. We show that the ability of RWs to find short paths bears no relation to the paths that they take. Instead, it relies on two properties of RWs on power law networks:

1) **RW's ability observe a sizable fraction of the network edges; and**
2) **an almost certainty that two distinct RW sample paths cross after a small percentage of the nodes have been visited.**

We show promising simulation results on several real world networks.

I. INTRODUCTION

Consider the following routing problem in the context of a large scale network G with $n \gg 1$ nodes, with particular interest paid to power law networks, although our results do not assume a particular degree distribution. A small set of nodes (h in total) want to exchange messages and are looking for short paths on G. We consider routing in two phases: Topology discovery and shortest paths calculation. In the topology discovery phase each node must crawl the network in order to partially discover the network topology. Nodes have limited crawling budget and only crawlers whose sample paths intersect are allowed to exchange topological information. Upon finishing the topology discovery phase nodes are allowed to optimally route on the sampled topology.

In this work consider crawling G with h independent random walkers (RWs). We show that the ability of RWs to find short paths bears no relation to the path that they take. Instead, it relies on two RW properties on power law networks:

- RW's ability observe a sizable fraction of the network edges; and
- two RW sample paths (starting at different nodes) cross after a small percentage of the nodes have been visited, with high probability.
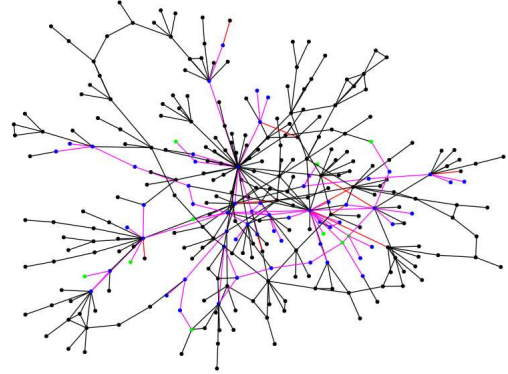

Fig. 1. Illustration of a RW sample path. Green nodes are the starting nodes, the blue nodes are nodes visited by the random walkers, and the purple edges are the edges used by the walkers to explore the graph which is a subset of $E(B,i)$.

We provide promising simulations results on several real world networks.

A. Framework

In what follows we formalize the model. Consider an undirected graph $G = (V, E)$ with n nodes and m edges. Start h random walkers from nodes $U = \{u_1, \ldots, u_h\}$ to explore the network. We assume each walker takes B steps. Each walker leaves bread crumbs so that one can trace its path back towards the initial node. Let $X(t, i) = (x_1, \ldots, x_t)$, $i = 1, \ldots, h$, denote the sequence of nodes seen by the i-th walker at step t. Let $S(t, i)$, $i = 1, \ldots, h$, denote the set of nodes visited by the i-th walker at time t, i.e., $S(t, i)$ is $X(t, i)$ without repeated nodes. And let $E(B, i)$ denote the set of edges belonging to the nodes visited by the i-th walker, $S(B, i)$, after the walker budget B is exhausted. Upon exausting its budget RW i outputs $X(B, i)$, $S(B, i)$, and $E(B, i)$.

B. Naive Routing

Note that if $S(B, i) \cap S(B, j) \neq \emptyset$, $i, j = 1, \ldots, h$, then walker i must have found walker j's breadcrumbs or vice-versa. Without loss of generality assume walker i found j's

Network	nodes	edges	$\langle k \rangle$	$\frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle}$
Flickr [12]	1.7M	31.1M	18.1	943.4
AS Net. [3]	26.5K	107K	4	279.24
Livejournal [12]	5.2M	9.8M	18.9	154
Enron [1]	37K	368K	10	140
Gnutella [15]	62.5K	296K	4.7	10.6
Power Grid [20]	5K	13K	2.7	2.9

TABLE I
SUMMARY OF NETWORKS

breadcrumbs. A naive way to construct a path between u_i and u_j is to route messages retracing all RW steps. Node u_i acomplishes this by retracing $X(B, i)$ to contact one of the nodes in $v \in S(B, i) \cap S(B, j)$. Once the message reaches node v, it reaches its final destination, node u_j, following j's breadcrumbs starting from v, i.e., the message follows the reverse RW path from u_j to v. The breadcrumb is installed only at the first visit to a node, which guarantees that the reverse RW path is without loops.

C. Drawbacks of Naive Routing

The naive algorithm presented above is quite inefficient. Random walks are not particularly good at finding short paths (as seen in Section V); unless there are few paths between pairs of nodes, e.g., G is a tree. A random walk with erased loops, also known as a loop-erased random walk, samples all possible spanning trees of G with equal probability [17]. This can lead to long paths lengths between u_i and u_j.

D. The True Power of RWs in Finding Short Paths

The ability of RWs to find short paths bears no relation to the paths that they take. Instead, it relies on RW properties found under conditions often present in power law networks. We summarize the main results of Section II here:

- $|E(B, i)| \approx \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle} B$, where $B \ll n$, $i = 1, \ldots, h$, and $\langle k \rangle$ and $\langle k^2 \rangle$ are the first and second moment of the node degrees in original graph G, respectively.
- Under mild conditions two RW sample paths cross, with high probability.

These observations motivate our use of RWs to partially uncover the topology of G.

E. Outline

The outline of this work is as follows. Section II characterizes the subgraph obtained by the i-th random walker and the probability that two RW sample paths cross. Section III presents the Random Walk Short Path (RWSP) algorithm. Section IV presents our results when simulating RWSP on real world networks. Finally, Section V presents a discussion and related work.

II. RW SUBGRAPH PROPERTIES

In this section we show that the real power of a random walk resides in its ability to collect edges from G. Let $E(B, i)$ denote the set of edges belonging to the nodes visited by the i-th walker, $S(B, i)$, after the walker budget B is exhausted.

We see that RW i, $i = 1, \ldots, h$, builds an edge rich subgraph $G'(B, i) = (V, E(B, \cdot))$ with budget $B = \beta n$ in a network with large degree variance. In what follows we assume $n \gg 1$, $B \ll n$, and that RW i has access to the neighbor list of each visited node, such that $E(B, i)$ is obtained without extra sampling costs.

These are the main results of this section:

- $|S(B, \cdot)| \approx B$, i.e., the set of nodes visited by the RW scales linearly with B. The RW tends to collect only new nodes at the beginning of the walk.
- $|E(B, i)| \approx \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle} B$, where $\langle k \rangle$ and $\langle k^2 \rangle$ are the first and second moment of the node degrees in original graph G, respectively.
- We provide conditions under which there is a RWSP path between any pair of nodes (u_i, u_j), $\forall i, j$, with high probability.

In power law networks $q = (\langle k^2 \rangle - \langle k \rangle)/\langle k \rangle$ is likely to be quite large, and thus G' spans a sizable fraction of G edges even when B is a small fraction of the total number of nodes. For instance, in the Flickr social photo sharing website $q = 943.4$, in the Enron email network this value is $q = 139$, and in the Livejournal blog posting social network this value is $q = 154$. Table I presents values of q for several real world networks.

A. The RW subgraph $G'(B, i)$ contains a sizable fraction of the edges of G.

In what follows we find a closed approximation formula that gives the number of edges in $G'(t, i)$, $t \leq B$. Let $S(t, i)$ denote the set of nodes visited by the i-th walker at time step t, $i = 1, \ldots, h$. As we are interested in large graphs, we assume $n \gg 1$. Let $\langle k \rangle$ and $\langle k^2 \rangle$ be the first and second moment of the node degrees in the original graph G, respectively. Let $\tau = t/n$ and $n_e(\tau) = E\left[|E(\tau n, i)|\right]$ be the average number of edges in $S(\tau n, i)$, where $E[\cdot]$ is the expectation operator. We provide a closed form mean field approximation (Theorem 2.1 in Section II-C) for $n_e(t)$ when $\tau \ll \langle k \rangle^2 / \langle k^2 \rangle$

$$n_e(\tau) = 2m \left(1 - \exp \left(-\frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle^2} \tau \right) \right). \qquad (1)$$

and

$$E[|S(\tau n, i)|] = \frac{n\langle k \rangle}{\langle k^2 \rangle - \langle k \rangle} \left(1 - \exp \left(-\frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle^2} \tau \right) \right).$$

Figures 2 and 3 compare our mean field approximation in (1) with simulation results for the Gnutella and Flickr networks. The Gnutella network cannot be considered a power law network as its maximum degree is quite small (one hundred), but our approximation works for any degree distribution not only for power laws. Please refer to Table I for characteristics of these two networks.

To see how $n_e(\tau)$ grows with $B = \beta n$ let's decompose (1) into its Taylor series expansion

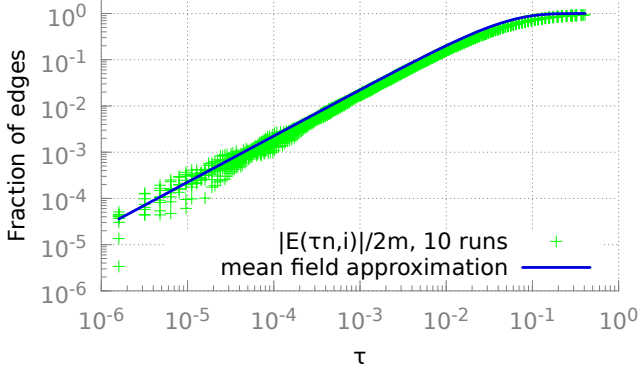$$n_e(\beta) = \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle} \beta n + O(\beta^2). \qquad (2)$$

Fig. 2. $n_e(\tau)$ **in the Gnutella network.** This graph shows the fraction of edges in $S(t, \cdot)$ against $\tau = t/n$ in the Gnutella network. The simulation results of 10 runs are shown in a scatter plot of red crosses. The mean field approximation in equation (1) is shown as a green line. In the Gnutella network $(\langle k^2 \rangle - \langle k \rangle)/\langle k \rangle = 10.6$, which is not as large as in a "true power law" network.
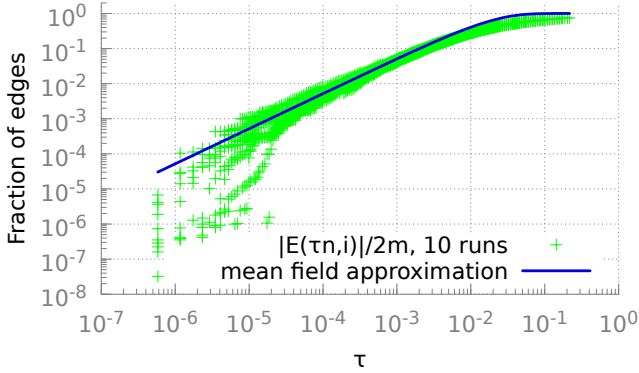

Fig. 3. $n_e(\tau)$ **in the Flickr network.** Same as Figure 2 over the Flickr network. In the Flickr network $(\langle k^2 \rangle - \langle k \rangle)/\langle k \rangle = 943.4$.

Eq. (2) yields for $\beta \ll 1$

$$n_e(\beta) \approx \frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle} \beta n \,. \qquad (3)$$

In a power law network in the asymptotic regime $n \to \infty$, i.e., if k is the node degree then it has degree distribution

$$p_k \propto (\alpha - 1)k^{-\alpha}. \qquad (4)$$

For power law degree distributions the approximation in (3) yields

$$n_e(\beta) = \begin{cases} \infty & \text{if } \alpha \leq 3 \\ \beta n \,(\alpha - 3)^{-1} & \text{if } \alpha > 3 \,. \end{cases}$$

While $n_e(\beta) \not\to \infty$ in any real world network, the above equation provides an invaluable hint that $n_e(\beta)$ can be quite large in real life. Indeed, in the Flickr network we observe

$$n_e^{\text{flickr}}(\beta) \approx 943.4 \times 1,700,000 \times \beta \text{ edges},$$

which means that by the end of the crawling the walker collects on average $943.4\beta n$ distinct edges. Table I shows the

value of $q = (\langle k^2 \rangle - \langle k \rangle)/\langle k \rangle$ for several real world networks. Before we proceed to the proof of equation (1), we switch gears and show that under certain conditions a RW starting at u_i, $i = 1, \ldots, h$, visits a node that is also visited by RW j, $\forall j \neq i$, with high probability.

B. With high probability two RW sample paths cross.

In what follows we show that under certain conditions the RW sample paths starting at nodes u_i and u_j, $\forall i, j$, cross with high probability. Assume, without loss of generality, that RW i completes before we start RW j. The intuition behind our result is that for sufficiently large $|E(B, i)|/2m$, $S(B, i)$ functions as an attractor for the j-th RW, such that with high probability RW j hits the set $S(B, i)$ before B steps. The above reasoning, however, cannot be applied to a RW that has a long transient; for instance a RW on a graph shaped like dumbbell likely will not satisfy our conditions.

We are interested in the hitting time of RW j into the set $S(B, i)$. Let $X = \{X_1^{(j)}, \ldots, X_B^{(j)}\}$ be the set of nodes visited by RW j. Then

$$P[X \cap S(B, i) = \emptyset]$$

denotes the probability that RW j and RW i do not meet. Unfortunately finding the hitting time to a set of nodes in a general graph is a hard problem. Solutions based on spectral methods (e.g., Chau and Basu [5, Theorem 1]) do not yield simple expressions for the general case. A more detailed discussion is found on Section V. Hence, we consider the following approximation. Let $B = \beta n$, $n \gg 1$, and

$$\gamma(B, i) = \frac{|E(B, i)|}{2m}$$

be the fraction of edges that belong to nodes in $S(B, i)$, i.e., nodes visited by the i-th RW. Note that $\gamma(B, i)$ is the steady state probability that RW j visits a node in $S(B, i)$ at any given time step. As $n \gg 1$ the average value of $\gamma(B, i)$ is

$$\bar{\gamma}(\beta) = E[\gamma(B, i)] = \frac{n_e(\beta)}{2m}.$$

Let $\Delta_n = o(n)$ and assume $\forall w \in \{1, B - \Delta\}$ that for some (possibly small) constant $c > 0$, the following inequality holds

$$P[X_{w+\Delta_n}^{(j)} \in S(\beta n, i)|X_w^{(j)} \notin S(\beta n, i)] \geq c\bar{\gamma}(\beta). \quad (5)$$

That is, if RW j is not at $S(B, i)$ at time w then at time $w + \Delta$ it will be at $S(B, i)$ with probability that is at least a constant factor of $\bar{\gamma}(\beta)$. In a graph with $o(n)$ mixing time $c \approx 1$ and the inequality in (5) becomes an equality. However, sublinear mixing time is a sufficient but not necessary condition in (5), which prompts us to conjecture that (5) holds for a broader class of graphs. Note that

$$P[X \cap S(B, i) = \emptyset] \leq P[\{X_1, X_{\Delta+1}, \ldots\} \cap S(B, i) = \emptyset]$$

as the probability that RW j at steps $1, \Delta + 1, \ldots$ does not hit $S(B, i)$ is higher than the probability that RW j at all steps $1, 2, \ldots, B$ does not hit $S(B, i)$. Application of (5) and the strong Markov property yields

$$P[X \cap S(B, i) = \emptyset] \leq (1 - c\bar{\gamma}(\beta))^{\lfloor \beta n/\Delta \rfloor}. \qquad (6)$$

From (1), $c\bar{\gamma}(\beta) > 0$. As $n \gg 1$, implies $\lfloor \beta n/\Delta \rfloor \gg 1$ we can put it all together into (6), yielding, for $\epsilon \ll 1$,

$$P[X \cap S(B, i) = \emptyset] \leq \epsilon.$$

Thus we conclude that if the condition states in (5) is satisfied then walkers i and j must meet with high probability. In what follows we provide a proof for (1).

C. Mean field analysis of the number of edges in $G'(t, i)$.

Theorem 2.1: Let $\tau = t/n \ll \langle k \rangle^2 / \langle k^2 \rangle$ and let $n_e(\tau) = E\left[|E(\tau n, i)|\right]$ be the average number of edges in $S(\tau n, \cdot)$. Then

$$n_e(\tau) = 2m \left(1 - \exp\left(-\frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle^2} \tau \right) \right).$$

Proof: By following an edge, a RW reaches a node v with degree k_v with probability proportional to k_v. Node v has average degree higher than the average degree in the graph. This is a classical case of the inspection paradox and therefore $E[k_v] = \langle k^2 \rangle / \langle k \rangle$. If $v \notin S(t, i)$ we are adding at most $k_v - 1$ edges to $E(t, i)$, (the minus one is a consequence of the need to remove the edge the RW followed to reach v which was already in $E(t, i)$). If G was a random configuration graph, as $t \ll n$ and $n \gg 1$, the probability that any of the $k_v - 1$ edges are already in $E(t, i)$ is negligible. Thus, by adding v we are adding $q = E[k_v] - 1$ edges to $E(t, i)$.

Assuming the RW is sampling edges uniformly at random, we can write the following expression for n_e

$$n_e(\tau + 1/n) - n_e(\tau) = qP[v \notin S(t, i)] = q \left(1 - \frac{n_e(\tau)}{2m} \right),$$

where $n_e(\tau)/2m$ is the probability that a randomly sampled edge is in $E(t, i)$. Multiplying both sides of the above equation by n and letting $n \to \infty$ yields

$$\frac{n_e(\tau + 1/n) - n_e(\tau)}{1/n} = nq \left(1 - \frac{n_e(\tau)}{2m} \right)$$

$$\frac{dn_e(\tau)}{d\tau} = \frac{q}{\langle k \rangle} (2m - n_e(\tau))$$

with the boundary condition $n_e(0) = 0$. The solution to the above differential equation is

$$n_e(\tau) = 2m(1 - e^{-q\tau/\langle k \rangle}),$$

concluding the proof. ∎

Lemma 2.2: Let $\tau = t/n \ll \langle k \rangle^2 / \langle k^2 \rangle$. Then

$$E[|S(\tau n, i)|] = \frac{n}{\langle k^2 \rangle - \langle k \rangle} \left(1 - \exp\left(-\frac{\langle k^2 \rangle - \langle k \rangle}{\langle k \rangle^2} \tau \right) \right).$$

Proof: The proof of follows from Lemma 2.1 with $E[|S(\tau n, i)|] = n_e(\tau)/q$. ∎

III. THE RANDOM WALK SHORT PATH (RWSP) ALGORITHM

The RWSP algorithm is as follows: Initialize $W(i) = \emptyset$ and $C(i) = \emptyset$, $i = 1, \ldots, h$.
1) Start walker i at node $u_i \in U$ with budget B, $i = 1, \ldots, h$.
2) For each walker $i = 1, \ldots, h$:
   a) At time step t update the "explored" graph $G'(t, i)$ as new nodes are visited and new edges are discovered.
   b) If the t-th visited node, x_t, has already been visited by a set of walkers $W \in \{1, \ldots, h\}^w$;
      i) Assign $W(i) \leftarrow W(i) \cup W$;
      ii) Assign $C(i) \leftarrow C(i) \cup \{x_t\}$;
      u_i sends a message to u_j advertising that node u_i just found x_t (sending this last message requires tracing back the path x_t to u_j using j's breadcrumbs), $\forall j \in W$.
   c) Stop when $t = B$.
3) At the end of walk send $G'(B, i)$ to u_j, $\forall j \in W(i)$, via the nodes in $C(i)$ that have been visited by walker j.
4) Upon receiving (j, v), $v \in V$, from walker $j \neq i$: assign $W(i) \leftarrow W(i) \cup \{j\}$ and $C(i) \leftarrow C(i) \cup \{v\}$.
5) Let $G^\star(i) = \bigcup_{r \in W(i) \cup \{i\}} G'(B, j)$ denote the union of all subgraphs known to u_i. Compute the minimum spanning tree $T(i)$ on $G^\star(i)$ starting from u_i.

Node u_i uses $T(i)$ to route its messages to nodes u_j on G, $\forall j \in W(i)$. Figure 1 shows a snapshot of the algorithm. In what follows we present our simulation results.

IV. RESULTS

This section presents our results when simulating RWSP on real world networks. Table I summarizes the datasets used in our simulations. Figure 4a show the fraction of RW shortest path lengths on $G^\star$ defined in Section III (X axis) between $h = 4$ nodes vs. the true shortest path lengths (Y axis) [blue/gray matrix]. We choose $h = 4$ walkers because as we increase k our results get better if B is kept constant. We also evaluate different values of h, $h \in \{2, 6, 8, 16\}$, rescaling B such that for a constant $\beta > 0$, $B = \beta n/h$. The results for different values of h are similar to those presented here, thus due to space constraints we omit them.

In Figure 4a the yellow bars next to the Y axis show the true all pairs shortest path length distribution. For instance, we in LiveJournal's most of the shortest paths are between 5 and 6 hops. The RW sampling budget is $B = 2.5\%$ of the nodes in the graph. A total of $10,000$ runs were used to produce the averages seen in the graph. The axis marking INF refers to the fraction of times that a node could not reach another node (due to disconnected components). In all runs the RWSP was able to find nodes in the same component (we condition the RWSP to start in the giant component). We also tested RWSP over other networks with similar parameters.

Our results are very promising. For Livejournal (Figure 4a) more than 60% of the RWSP paths of any given length are the
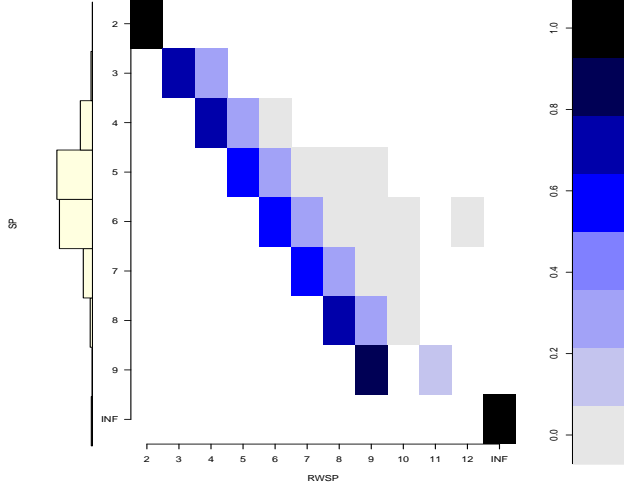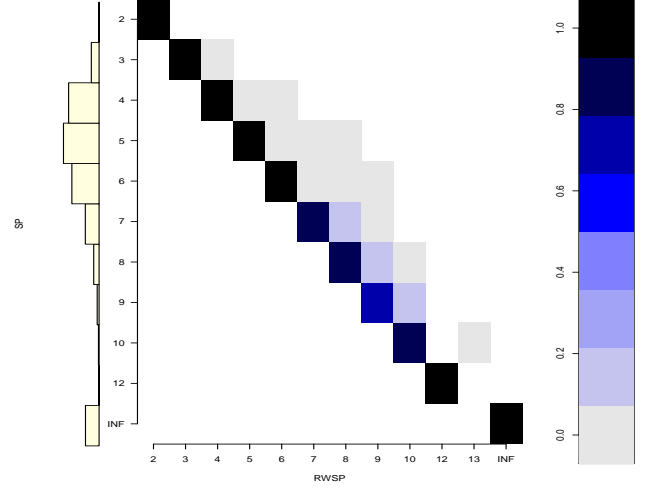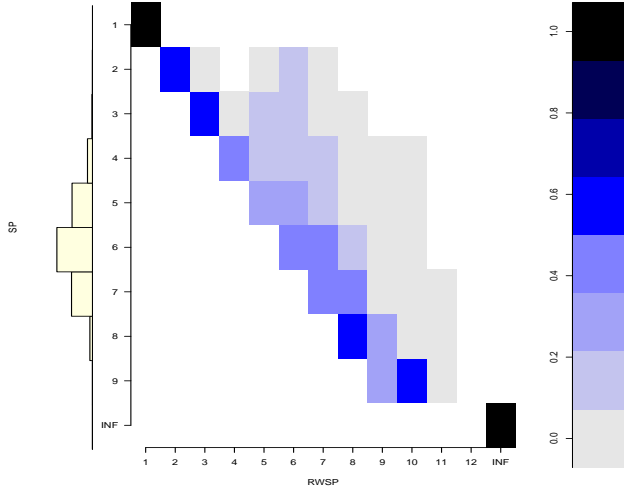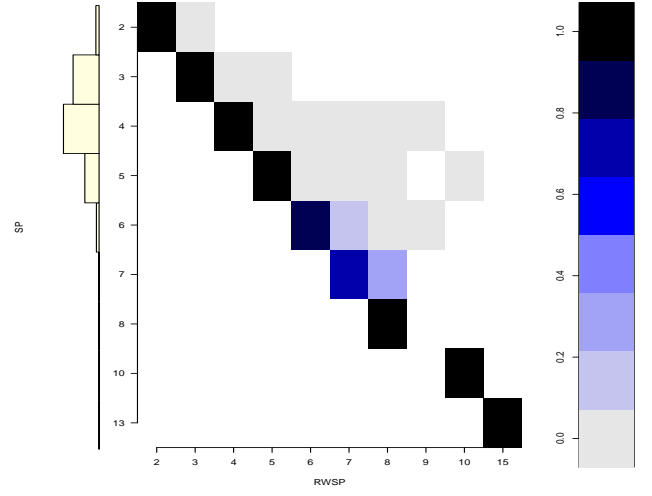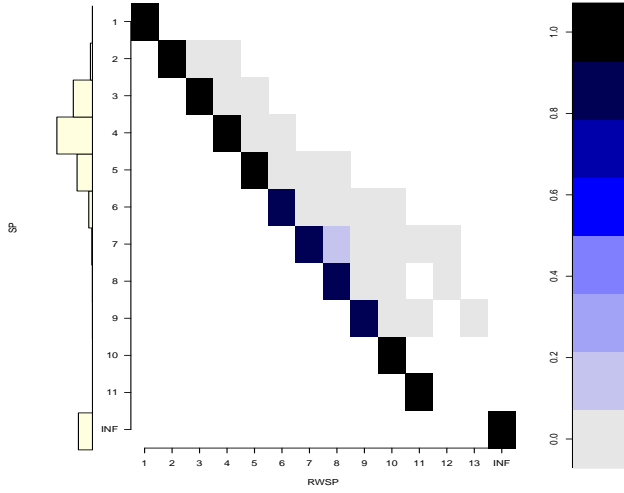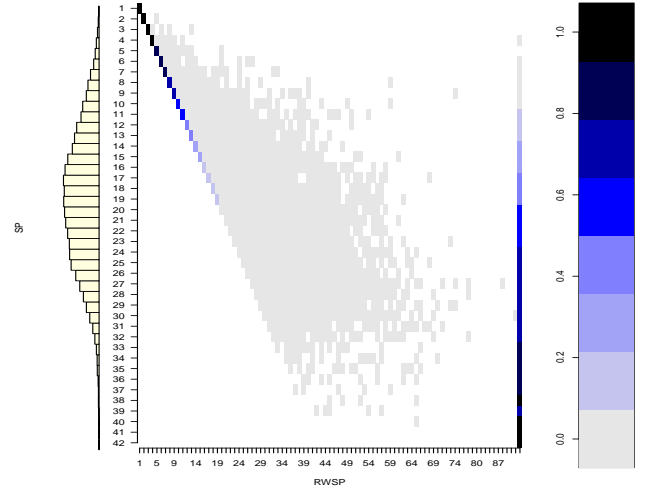
(a) LiveJournal ($h = 4$, $B = 0.025\,n$)

(b) Flickr ($h = 4$, $B = 0.025\,n$)

(c) Gnutella ($h = 4$, $B = 0.05\,n$)

(d) AS graph ($h = 4$, $B = 0.0125\,n$)

(e) Enron ($h = 16$, $B = 0.0125\,n$)

(f) Power Grid ($h = 4$, $B = 0.05\,n$)

Fig. 4. **RWSP vs. Shortest Paths.** h walkers and sampling budget B. Fraction of RW shortest path lengths (X axis) between nodes vs. true shortest path lengths (Y axis) [blue/gray matrix]. Yellow bars next to Y axis show true shortest path lengths distribution. The axis marking INF refers the fraction of times that a node could not reach another node (due to disconnected components).

shortest paths. Moreover, more than 90% of the RWSP paths of any given length are within one hop of the shortest paths.

We also test RWSP on Flickr, AS graph, a snapshot of the Gnutella network, Enron email dataset with similar results. On Flickr (Figure 4b) ($B = 0.0125\,n$) more than 85% of the RWSP paths of any given length are the shortest paths; also, all paths are within 2 hops of the shortest paths. On Gnutella (Figure 4c) ($B = 0.0125\,n$) network we observe more than 60% of RWSP paths of all lengths to be within 2 hops of the shortest paths. For the AS graph (Figure 4d) (RWSP with $B = 0.05\,n$), more than 65% of RWSP paths of all lengths are the shortest paths; the majority ($> 95\%$) of RWSP paths now counting over all lengths are the shortest paths. On Enron (Figure 4e) ($B = 0.05\,n$) we have a result as good as the one for the AS graph.

Figure 4f shows that the only graph in which the RWSP does not perform well, the Power Grid network. The Power Grid network is not a power law graph and has the largest diameter of any of the previous networks, thus, we expected RWSP to perform poorly. Consulting Table I we observe that $(\langle k^2 \rangle - \langle k \rangle)/(\langle k \rangle) = 2.9$ is small and thus G' spans only a small fraction of the original graph, making the task of finding a long path close to impossible.

V. Discussion & Related Work

The problem of topology discovery has received significant attention in the (communications) networking literature. Broadly, there are two types of topology discovery – *proactive* and *on-demand*. In proactive "link-state" routing schemes such as OLSR [7], each node *floods* the state of its neighborhood to the entire network. Since each flood costs $O(m)$ forwardings, where a network has n nodes and $m = O(n^2)$ edges, the total overhead of flooding the network is $O(mn)$ which is $O(n^3)$ for dense networks. Variants of link-state routing such as HSLS [16] reduce this overhead in dynamic networks to $O(n^{5/2})$ (amortized over time) by restricting the frequency of flooding the link-state updates to nodes located farther away; or they optimize the flooding proceeding by using Multi Point Relays [9]. These schemes are advantageous when the likely traffic patterns are *all-to-all*, i.e., every node is expected to transmit; hence, the cost of topology updates is amortized over all the nodes.

However, often only a small number h of nodes in the network need to transmit or receive messages to/from each other. In these situations, on-demand routing schemes are preferred since they attempt to gather topology state when they need to communicate [8], [13]. This is typically achieved by means of flooding the network with a "route request" message. When the intended recipient receives this message, it replies with a "route response" message along the reverse path to the source (this is unicast). In the worst case, this takes $hO(m)$ forwardings in dense networks.

Opportunistic routing schemes, in contrast, do not attempt to discover the topology but rather make routing decisions adaptively based on the actual transmission outcomes. The back-pressure algorithm pioneered by Tassiulas

and Ephremides [19] explores and exploits all feasible paths between each source and destination to achieve throughput-optimal routes. Ant colony optimization is a meta-heuristic technique that uses artificial ants (specially marked packets) to build pheromone trails in the network [4].

Random walks have also been used to deliver messages but suffer from excessive delays due to long hitting times [18], [11]. Hitting times of random walks have been well studied. For a large $K \times K$ torus, as $K \to \infty$, the maximum hitting time between two nodes is $H_{max} = \Theta(K^2 \log K)$. We showed in our previous work [6] that for network topologies with node degrees concentrated around the mean r, $H_{max} \approx \frac{n \log n}{r^2}$. A corollary to the above is applicable to random geometric networks at the critical radius [14]; in that regime, $r \approx \log n$, and therefore, $H_{max} = O(n)$. Although in the worst case hitting times could have a $O(n^3)$ scaling law [10] (for "dumbbell" shaped topologies), in most commonly occurring topologies, the scaling properties of H_{max} reside between $O(n)$ and $O(n^2)$. Moreover, the hitting time to a *subgraph* drops sharply when the size of the subgraph increases in comparison to the hitting time to a specific node [2]. Thus, this could yield lower overhead than on demand protocols described above, albeit at the cost of a higher routing stretch (the on demand routing protocols return optimal shortest paths).

VI. Acknoledgements

References

[1] Y. Yang B. Klimmt. Introducing the enron corpus. In *CEAS conference*, 2004.

[2] Prithwish Basu and Saikat Guha. Effect of limited topology knowledge on opportunistic forwarding in ad hoc wireless networks. In *WiOpt*, pages 237–246. IEEE, 2010.

[3] Caida as graph from november 5 2007.

[4] G. Di Caro and M. Dorigo. Antnet: Distributed stigmergetic control for communications networks. *J. of Artificial Intelligence Research*, 9:317–365, 1998.

[5] Chi-Kin Chau and P. Basu. Exact analysis of latency of stateless opportunistic forwarding. In *IEEE INFOCOM*, pages 828–836, 2009.

[6] Chi-Kin Chau and P. Basu. Analysis of latency of stateless opportunistic forwarding in intermittently connected networks. *IEEE/ACM Transactions on Networking*, 19(4):1111–1124, 2011.

[7] Thomas Clausen, Philippe Jacquet, Anis Laouiti, Paul Muhlethaler, Amir Qayyum, and Laurent Viennot. Optimized link state routing protocol for ad hoc networks. In *IEEE INMIC Pakistan*, 2001.

[8] David B. Johnson and David A. Maltz. Dynamic Source Routing in Ad Hoc Wireless Networks. In Imielinski and Korth, editors, *Mobile Computing*, volume 353. Kluwer, 1996.

[9] Anis Laouiti, Amir Qayyum, and Laurent Viennot. Multipoint relaying: An efficient technique for flooding in mobile wireless networks. In *35th Annual Hawaii International Conference on System Sciences (HICSS'2001)*, Maui, 2001. IEEE Computer Society.

[10] L. Lovasz. Random walk on graphs: A survey. *Combinatorics, Paul Erdos is Eighty*, 2:1–46, 1993.

[11] Issam Mabrouki, Xavier Lagrange, and Gwillerm Froc. Random walk based routing protocol for wireless sensor networks. In *Proc. of the 2nd ValueTools*, pages 71:1–71:10, 2007.

[12] Alan Mislove, Massimiliano Marcon, Krishna P. Gummadi, Peter Druschel, and Bobby Bhattacharjee. Measurement and Analysis of Online Social Networks. In *Proc. of the IMC*, October 2007.

[13] C. Perkins, E. Royer, and S. Das. RFC 3561 Ad hoc On-Demand Distance Vector (AODV) Routing. Technical report, IETF, 2003.

[14] Sanatan Rai. The spectrum of a random geometric graph is concentrated. *Journal of Theoretical Probability*, 20(2):119–132, 2007.

[15] M. Ripeanu, I. Foster, and A. Iamnitchi. Mapping the gnutella network: Properties of large-scale peer-to-peer systems and implications for system design. *IEEE Internet Computing Journal*, 2002.

[16] César A. Santiváñez, Ram Ramanathan, and Ioannis Stavrakakis. Making link-state routing scale for ad hoc networks. In *Proc. Mobihoc*, pages 22–32, 2001.

[17] Oded Schramm. Scaling limits of loop-erased random walks and uniform spanning trees. *Israel J. Math.*, 118:221–288, 2000.

[18] Sergio D. Servetto and Guillermo Barrenechea. Constrained random walks on random graphs: routing algorithms for large scale wireless sensor networks. In *Proc. of the WSNA*, pages 12–21, 2002.

[19] L. Tassiulas and A. Ephremides. Stability properties of constrained queueing systems and scheduling policies for maximum throughput in multihop radio networks. *IEEE Transactions on Automatic Control*, 37(12):1936–1948, 1992.

[20] D. J. Watts and S. H. Strogatz. Collective dynamics of small-world networks. *Nature*, 393:440–442, 1998.